

1

《C/C++ 学习 指南》

第13.3讲：链表中插入与删除节点

作者：邵发 QQ群：417024631

官网：<http://www.afanihao.cn/>

答疑：<http://www.afanihao.cn/kbase/>

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

2

版权所有，侵权必究

按顺序插入节点

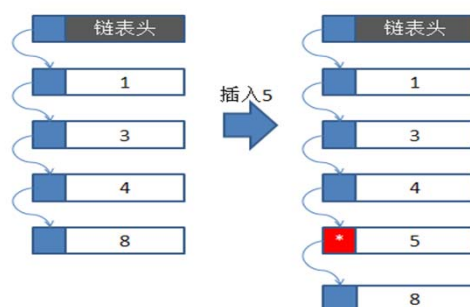
第13.2中已经介绍了两种添加节点的方法：添加到最前面、或者添加到末尾。

这里再介绍一种按排序插入的情况。例如，要求链表中的Student对象的id按从小到大顺序。

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

按顺序插入节点

原链表中已经存在1, 3, 4, 8四个节点，新入一个ID为5的节点，如下图所示：



由于每个节点插入的时候，都是按顺序插入的，所以最终这个链表是按顺序排列的。

《C/C++学习指南》邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

按顺序插入节点

如何实现按顺序插入？

方法：在插入时，遍历链表，并比较ID的值，找到目标位置。

注：链表插入的核心操作，是找到目标位置，并记录前一个节点pre。

新节点直接挂在pre后面就行了，这个操作很简单。

```
obj->next = pre->next;
```

```
pre->next = obj;
```

《C/C++学习指南》邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

按顺序插入节点

```
// 按顺序插入节点
int insert(Student* obj)
{
    Student* cur = &m_head.next; // 当前节点current
    Student* pre = &m_head; // 上一个节点previous
    while(cur)
    {
        if(obj->id < cur->id) // 找到这个位置
            break;
        pre = cur;
        cur = cur->next; // 找到最后一个对象
    }

    // 插入到pre节点的后面
    obj->next = pre->next;
    pre->next = obj;
    return 0;
}
```

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

查找和删除节点

删除节点的核心目标：找到这个节点，并记录该节点的前一个节点pre。

删除：

```
pre->next = obj->next;
free(obj);
```

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

查找和删除节点

```
// 按id查找并删除节点
void remove(int id)
{
    Student* cur = m_head.next; // 当前节点current
    Student* pre = &m_head; // 上一个节点previous
    while(cur)
    {
        if(id == cur->id) // 找到这个位置
        {
            // 删除该节点
            pre->next = cur->next;
            free(cur);
            break;
        }
        pre = cur;
        cur = cur->next; // 找到最后一个对象
    }
}
```

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

与数组比较

数组：如果要在中间插入/删除一个对象，那么要把后面所有的元素后移，花费的cpu时间较多

链表：在中间插入/删除一个对象，不需要数据移动，直接挂中“链条”中即可。

所以，链表比数组适合插入/删除操作，性能较高。

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

与“无头链表”比较

为什么不使用“无头链表”？

因为“无头链表”的插入和删除都比较复杂。当要操作的节点位于头部时，非常复杂。

“有头节点”只是增加了一个固定的头部，在内存空间上的浪费是可以忽略不计的。但是带来的方便却是非常地大。

小结

介绍了在“有头链表”中插入和删除节点的方法。